07-19-04 10/788,417
KUMAGAI et al.
March 1, 2004

日本国特許庁
JAPAN PATENT OFFICE

McDermott Will & Emery LLP

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日
Date of Application: ２００３年１０月１０日

出願番号
Application Number: 特願２００３－３５１５０９

[ST. 10/C]: [ＪＰ２００３－３５１５０９]

出願人
Applicant(s): 株式会社日立製作所

２００４年 ３月 １日

特許庁長官
Commissioner,
Japan Patent Office

今井康夫

出証番号 出証特２００４－３０１４９５８

【書類名】　特許願
【整理番号】　K03010131A
【あて先】　特許庁長官殿
【国際特許分類】　H04L　9/00
【発明者】
　【住所又は居所】　神奈川県川崎市麻生区王禅寺１０９９番地　株式会社日立製作所システム開発研究所内
　【氏名】　熊谷　洋子
【発明者】
　【住所又は居所】　神奈川県川崎市麻生区王禅寺１０９９番地　株式会社日立製作所システム開発研究所内
　【氏名】　藤城　孝宏
【発明者】
　【住所又は居所】　神奈川県川崎市麻生区王禅寺１０９９番地　株式会社日立製作所システム開発研究所内
　【氏名】　鍛　忠司
【発明者】
　【住所又は居所】　神奈川県川崎市麻生区王禅寺１０９９番地　株式会社日立製作所システム開発研究所内
　【氏名】　羽根　慎吾
【発明者】
　【住所又は居所】　東京都江東区新砂一丁目６番２７号　株式会社日立製作所公共システム事業部内
　【氏名】　下之薗　仁
【特許出願人】
　【識別番号】　000005108
　【氏名又は名称】　株式会社　日立製作所
【代理人】
　【識別番号】　100075096
　【弁理士】
　【氏名又は名称】　作田　康夫
【手数料の表示】
　【予納台帳番号】　013088
　【納付金額】　21,000円
【提出物件の目録】
　【物件名】　特許請求の範囲　1
　【物件名】　明細書　1
　【物件名】　図面　1
　【物件名】　要約書　1

【書類名】特許請求の範囲
【請求項１】
証明書の有効性確認局装置において、依頼に応じて行う、公開鍵証明書の有効性確認方法であって、
予め、パスの検索と、検索したパスの検証と、を行うステップと、
前記検索と検証の結果を、予め定めた基準に基づき、前記パスを分類して、データベースに登録する、パス登録ステップと、
端末装置から公開鍵証明書の有効性確認依頼を受け取り、予め登録されたパスを用いて検証する、有効性確認ステップと、を有する。
【請求項２】
請求項１記載の公開鍵証明書の有効性確認方法であって、
前記有効性確認ステップにおいて、当該有効性確認依頼に対応する有効なパスが登録されていない場合、新たにパス検索、および検証をすることにより、前記公開鍵証明書の有効性確認を行う。
【請求項３】
請求項１記載の公開鍵証明書の有効性確認方法であって、
前記パス登録ステップにおいて、前記予め定めた基準とは、検証結果に応じて、有効なパスか、有効でないパスかであり、
前記有効性確認ステップにおいて、当該有効性確認依頼に対応するパスが、有効なパス、または有効でないパスとして、前記データベースに登録されている場合は、前記登録結果に従って、前記依頼された公開鍵証明書の有効性確認を行う。
【請求項４】
請求項３記載の公開鍵証明書の有効性確認方法であって、
前記有効性確認ステップにおいて、当該有効性確認依頼に対応するパスが、有効なパスとして登録されている場合でも、当該公開鍵証明書または当該パスに含まれる公開鍵証明書に、制限事項が記述されている場合は、前記有効性確認依頼に応じてパス検証を行い、当該公開鍵証明書および当該パスが前記制限事項を満たしているかどうかを調べるステップと、
制限事項を満たしていれば、有効なパスと判断するステップと、を有する。
【請求項５】
請求項３記載の公開鍵証明書の有効性確認方法であって、
前記有効性確認ステップにおいて、当該有効性確認依頼に対応するパスが、有効なパスとして登録されている場合でも、当該有効性確認依頼または当該公開鍵証明書または当該パスに含まれる公開鍵証明書に、ポリシーが記載されている場合は、前記有効性確認依頼に応じてパス検証を行い、当該公開鍵証明書および当該パスが、電子手続きのポリシーを満たしているかどうかを調べるステップと、
前記ポリシーを満たしている場合に、有効なパスと判断するステップと、を有する。
【請求項６】
請求項３記載の、公開鍵証明書の有効性確認方法であって、
前記パス登録ステップは、トラストアンカー認証局から、エンドエンティティ証明書を発行する認証局までのパスを検索するステップと、
当該エンドエンティティ証明書を発行する認証局が発行する、エンドエンティティ証明書に関する失効リストを取得、検証するステップと、
当該失効リストの検証結果とともに、失効リストを登録するステップと、を有する。
【請求項７】
請求項６記載の公開鍵証明書の有効性確認方法であって、
前記有効性確認ステップにおいて、当該有効性確認依頼に対応するパスが、前記データベースに有効なパスとして登録されている場合、前記失効リストの検証を行わずに、当該公開鍵証明書が失効していないことを確認する。

【書類名】明細書
【発明の名称】公開鍵証明書検証の高速化方法、および装置
【技術分野】
【０００１】
本発明は、公開鍵基盤（Ｐｕｂｌｉｃ　Ｋｅｙ　Ｉｎｆｒａｓｔｒｕｃｔｕｒｅ：以下ＰＫＩという）において、ある端末が受け取った電子手続に対する署名を検証するための公開鍵証明書に関して、その有効性を確認するのに好適な技術に関する。
【背景技術】
【０００２】
民間系、公共系の様々な組織、団体において、従来、書面で行ってきた様々な手続を電子化するべく、ＰＫＩの導入、整備が進んでいる。
【０００３】
図１２は、ＰＫＩにおいて、認証局（Ｃｅｒｔｉｆｉｃａｔｅ　Ａｕｔｈｏｒｉｔｙ：以下ＣＡという）が複数ある場合における各ＣＡの関係の例を示している。
【０００４】
図示するように、公開鍵証明書の発行とその管理を行う各ＣＡは、ルートＣＡ１を頂点とするツリー構造を持つグループを形成している。このグループはセキュリティドメインと呼ばれており、１つのポリシー管理機関の基で運用されるＰＫＩの単位である。ルートＣＡ１は、自身より１つ下流側に位置する各$ＣＡ２_{1}$～$ＣＡ２_{n}$に対して公開鍵証明書を発行する。また、$ＣＡ２_{1}$～$ＣＡ２_{n}$各々は、自身より１つ下位に位置する各$ＣＡ３_{1}$～$ＣＡ３_{n\ 1}$に対して公開鍵証明書を発行する。このように、ツリー上、１つ上位に位置するＣＡが自身より１つ下位側に位置するＣＡに対して公開鍵証明書を発行する。そして、ツリー上、最下位に位置するＣＡ（以下、エンドエンティティ証明書発行ＣＡと呼ぶ）$ＣＡｓ_{1}$～$ＣＡｓ_{n\ m}$は、電子手続を行うユーザ（以下、Ｅｎｄ　Ｅｎｔｉｔｙ：ＥＥと呼ぶ）$ＥＥ_{1}$～$ＥＥ_{X}$に対して公開鍵証明書を発行する。
【０００５】
$ＥＥ_{1}$～$ＥＥ_{X}$各々の装置が電子文書の署名を生成する際に使用する秘密鍵（署名鍵）の正当性は、自身を収容する端末収容認証局$ＣＡｓ_{1}$～$ＣＡｓ_{n\ m}$が発行した公開鍵証明書によって証明され、端末収容認証局$ＣＡｓ_{1}$～$ＣＡｓ_{n\ m}$各々の装置が発行する公開鍵証明書の署名を生成する際に使用する秘密鍵の正当性は、自身を収容する認証局$ＣＡ（ｓ－１）_{1}$～$ＣＡ（ｓ－１）_{n\ (m-1)}$が発行した公開鍵証明書によって証明される。したがって、$ＥＥ_{1}$～$ＥＥ_{X}$各々の装置が署名を生成する際に使用する秘密鍵は、最終的に、ルート認証局ＣＡ１が発行する公開鍵証明書によって証明される。この$ＥＥ_{1}$～$ＥＥ_{X}$の装置が署名を生成する際に使用する鍵の正当性を最終的に証明する認証局、云いかえれば、$ＥＥ_{1}$～$ＥＥ_{X}$が信頼する、ツリー上、最上位のＣＡをトラストアンカーＣＡと呼ぶ。
【０００６】
さて、図１２において、$ＥＥ_{1}$装置は、$ＥＥ_{X}$装置に送信すべき電子文書に対して、$ＥＥ_{1}$が保持する自身の秘密鍵で署名を行う。そして、署名した電子文書に、$ＣＡｓ_{1}$が発行した前記秘密鍵と対の$ＥＥ_{1}$の公開鍵証明書を添付して、$ＥＥ_{X}$装置に送信する。
【０００７】
$ＥＥ_{X}$装置は、$ＥＥ_{1}$装置から受け取った電子文書の署名を、当該電子文書に添付された$ＥＥ_{1}$の公開鍵証明書を用いて検証することができる。しかし、$ＥＥ_{1}$の公開鍵証明書は$ＥＥ_{X}$の証明書発行$ＣＡｓ_{n\ m}$が発行したものではないので、$ＥＥ_{X}$は、自身のトラストアンカーＣＡであるルートＣＡ１によって当該公開鍵証明書の有効性が証明されるものであることを確認しなければ、当該公開鍵証明書を信頼することはできない。この公開鍵証明書の有効性確認処理は、以下の手順によって行われる。
（１）トラストアンカーＣＡから公開鍵証明書の発行元ＣＡまでのパスの検索
トラストアンカーＣＡ（ここでは、ルートＣＡ１）を起点ＣＡとし、起点ＣＡが発行した公開鍵証明書の発行先を調べ、当該発行先にＣＡが含まれる場合は当該ＣＡが発行した

公開鍵証明書の発行先をさらに調べる処理を、当該公開鍵証明書の発行先に、ＥＥ証明書発行ＣＡ（ここでは、EE_1の公開鍵証明書を発行するＣＡｓ１）が含まれるまで続け、トラストアンカーＣＡからＥＥ証明書発行ＣＡまでのパスを検索する。

（２）検索したパスの検証

上記（１）により検索したパス上の各ＣＡから、パス上において当該ＣＡの１つ下位側のＣＡに対して発行した公開鍵証明書を入手する。そして、有効性確認対象の公開鍵証明書（ここでは、ＥＥ証明書発行CAs_1がEE_1に対して発行した公開鍵証明書）の署名を、当該公開鍵証明書を発行したＣＡ（ここでは、ＥＥ証明書発行CAs_1）より１つ上位のＣＡが発行した公開鍵証明書で検証し、検証できた場合は、当該１つ上位のＣＡが発行した公開鍵証明書の署名をさらに１つ上位のＣＡが発行した公開鍵証明書で検証する処理を、当該１つ上位のＣＡがトラストアンカーＣＡとなるまで続ける。そして、このような公開鍵証明書の署名検証がトラストアンカーまでできた場合に、有効性確認対象の公開鍵証明書の有効性が確認されたものとする。

【０００８】

EE_x装置は、EE_1装置より受け取った電子文書の署名を当該電子文書に添付されたEE_1の公開鍵証明書を用いて検証すると共に、上記の（１）、（２）に示す手順に従い当該電子文書の署名検証に用いたEE_1の公開鍵証明書の有効性を確認することにより、当該電子文書の正当性を確認することができる。

【０００９】

なお、以上では、公開鍵証明書の有効性確認処理をＥＥ装置で行うことを前提としている。しかし、公開鍵証明書の有効性確認処理は負荷が重く、これをＥＥで行うためには、当該ＥＥ装置に高い処理能力が要求される。そこで、ネットワークを介してＥＥ装置に接続された証明書の有効性確認局（以下、Ｖａｌｉｄａｔｉｏｎ　Ａｕｔｈｏｒｉｔｙ：ＶＡという）を設け、当該ＶＡ装置に、当該ＥＥの代わりに公開鍵証明書の有効性確認を行わせることが、インターネットに関するさまざまな技術の標準化を定める団体であるＩＥＴＦ（Ｉｎｔｅｒｎｅｔ　Ｒｅｓｅａｒｃｈ　Ｔａｓｋ　Ｆｏｒｃｅ）により提案されている。ＶＡ装置において、公開鍵証明書の有効性確認を行う場合、まずＥＥ装置は公開鍵証明書の有効性確認依頼を、ＶＡ装置に送付する。次に、ＶＡ装置において、上記（１）、（２）の処理を行い、最後に、その結果をＥＥ装置に送付する。

【００１０】

このとき、ＥＥ装置が公開鍵証明書の有効性確認を依頼してから、その結果が分かるまでにかかる時間を短くするための方法として、次のようなものがある。

【００１１】

ＶＡ装置において、予め定期的にパスを検索し、パスデータベースに登録しておく。そして、あるＥＥ装置から公開鍵証明書の有効性確認の依頼があった場合に、ＶＡ装置のパスデータベースに、対応するパスを検索し、検索したパスの検証を行うことにより、前記公開鍵証明書の有効性を確認する（例えば特許文献１を参照）。

【００１２】

また、もう１つの方法では、ＶＡ装置において、予め定期的に全てのパスを検索し、検索したパスの検証を行う。検証が成功したパス（有効なパス）についてのみ、パスデータベースに登録しておく。そして、あるＥＥ装置から公開鍵証明書の有効性確認の依頼があった場合に、ＶＡ装置のパスデータベースに、対応するパスが登録されているか否かを調べることで、前記公開鍵証明書の有効性を確認する（例えば特許文献２を参照）。

【００１３】

【特許文献１】米国特許第６１３４５５０号明細書

【００１４】

【特許文献２】米国特許出願公開第２００２／０４６３４０号明細書

【発明の開示】

【発明が解決しようとする課題】

【００１５】

上記特許文献２に記載されていた方法では、ＥＥ装置から受け取った公開鍵証明書の有効性確認依頼に対応するパスが、パスデータベースに登録されていない場合は、有効性確認対象証明書が有効でないものと判断していた。しかし、この方法に従うと、パス検索時に存在しなかったパスが、ＥＥ装置からの公開鍵証明書の有効性確認依頼を受け取った時に、新たに存在する場合には、有効な公開鍵証明書を有効でないと判断してしまうことがある。上記特許文献２には、このような場合の処理については記述されていなかった。

【００１６】

また、上記特許文献１にも、ＥＥ装置から受け取った公開鍵証明書の有効性確認依頼に対応するパスが、パスデータベースに登録されていない場合の処理については記述されていなかった。

【００１７】

このような、ＶＡで受け付けた有効性確認依頼に対応するパスがパスデータベースに登録されていない場合に、新たにパス検索、検証を行うことで適切な結果を応答することができる。しかし、この場合の有効性確認処理時間が長くなってしまうという課題がある。

【００１８】

現在、民間系、公共系のさまざまな組織、団体において、ＰＫＩの導入、整備が進んでおり、その結果、多数のセキュリティドメインが並列し、複雑なＰＫＩ構成となることが予想される。さらに、ＰＫＩを利用したアプリケーションが普及すると、たくさんの有効性確認依頼がなされると予想される。このような場合、ＥＥが公開鍵証明書の有効性確認を依頼してから、その結果が分かるまでにかかる時間が長くなり、サービスの低下を招いてしまう。

【課題を解決するための手段】

【００１９】

本発明は、パス検索時に存在しなかったパスが、パス検索時より後に形成された場合でも適切な結果を応答する技術、および／または、ＥＥが公開鍵証明書の有効性確認を依頼してからその結果が分かるまでにかかる時間をさらに短くする技術を提供する。

【００２０】

具体的には、本発明では、ネットワークを介して、複数の端末装置（ＥＥ装置）やＣＡ装置に接続されたＶＡ装置は、あるＥＥ装置からの依頼に応じて、公開鍵証明書の有効性を確認するために以下の処理を行う。

【００２１】

公開鍵証明書を発行することにより関係付けられた全てのＣＡに関して、存在するパスを全て検索し、前記パス検索により検出されたパスについて検証を行う。また、検出したパスの端点に位置するＥＥ証明書発行ＣＡが発行する、ＥＥ証明書に関する失効リスト（Ｃｅｒｔｉｆｉｃａｔｅ　Ｒｅｖｏｃａｔｉｏｎ　Ｌｉｓｔ，以下ＣＲＬという）を取得する。取得したＣＲＬについて、当該ＣＲＬが有効期間内であることを確認するとともに、当該ＣＲＬの発行ＣＡの公開鍵証明書により、当該ＣＲＬの署名検証を行う。そして、検証できたパスと、検証できなかったパスを、分類して、パスデータベースに登録する。このパスデータベース作成処理は、ＥＥからの公開鍵証明書の有効性確認依頼とは独立して、所定の規則に従って、繰り返し、例えば定期的に行う。

【００２２】

そして、あるＥＥから、公開鍵証明書の有効性確認依頼があった場合に、それに対応するパスが、パスデータベースに登録されているか否かを調べ、検証できたパスとしてデータベースに登録されている場合は、データベースに登録されたＣＲＬを用いて、当該公開鍵証明書が失効しているか否かを確認することにより、当該公開鍵証明書の有効性を確認する。

【００２３】

一方、当該公開鍵証明書に対応するパスが、検証できなかったパスとしてパスデータベースに登録されている場合は、当該登録されている有効でないパス以外に、有効なパスが存在するかどうかを調べ、存在しない場合には、当該公開鍵証明書は検証できなかったも

のとする。新たなパスやＣＲＬを検出した場合には、当該パスや当該ＣＲＬを用いて当該公開鍵証明書の有効性を確認し、当該検証結果をもとに、パスデータベースに追加登録を行う。

【００２４】

また、有効性確認依頼があった公開鍵証明書に対応するパスやＣＲＬが、パスデータベースに登録されていない場合は、新たにパスやＣＲＬの検索、検証処理を行うことにより、上記公開鍵証明書の有効性を確認する。このとき、新たなパスやＣＲＬを検出した場合には、当該パスや当該ＣＲＬの検証結果をもとに、パスデータベースに追加登録を行う。

【００２５】

本発明によれば、あるＥＥから公開鍵証明書の有効性確認依頼を受けた場合に、パス検索時より後に新たなパスが形成された場合でも、適切な結果を応答することができる。かつ、有効性確認依頼に対応するパスが、検証できたパスとして登録されている場合には、上記の（１）、（２）に示した、当該ＥＥのトラストアンカーＣＡから当該公開鍵証明書のＥＥ証明書発行ＣＡまでのパスの検索、検出したパスの検証、および、当該公開鍵証明書に対応するＣＲＬの署名検証を行う必要がない。また、有効性確認依頼に対応するパスが、検証できなかったパスとして登録されている場合には、パス検索、検証を、少ない処理で行うことができる。したがって、あるＥＥが、公開鍵証明書の有効性確認を依頼してから、当該有効性が確認されるまでにかかる時間を短縮できる。

【発明の効果】

【００２６】

本発明によれば、パス検索時より後に新たなパスが形成された場合でも、適切な結果を応答することができ、および／または、ＥＥが公開鍵証明書の有効性確認を依頼してからその結果が分かるまでにかかる時間を短縮することができる。

【発明を実施するための最良の形態】

【００２７】

図１は、本発明の一実施形態が適用されたＰＫＩシステムの概略構成を示す図である。

【００２８】

本実施例のＰＫＩシステムは、電子的に手続を行う複数のＥＥ$_1$装置（１１）～ＥＥ$_N$装置（１１）（ＥＥ装置（１１）と総称する）と、公開鍵証明書発行業務を行うＣＡ$_1$装置（１３）～ＣＡ$_M$装置（１３）と、公開鍵証明書の有効性確認局ＶＡ（１４）と、それぞれを接続するインターネット等のネットワーク（以下、ＮＥＴという）１６からなる。

【００２９】

図２は、図１に示すＰＫＩシステムでの各ＣＡの関係の一例を示す図である。

【００３０】

図示するように、本実施形態のＰＫＩシステムでは、民間系、政府系といった複数のセキュリティドメインＳＤ（ＳＤ１～ＳＤ２）が並存していることを前提としている。また、各セキュリティドメインＳＤのルートＣＡ（図２ではＣＡ$_{11}$、ＣＡ$_{21}$、）は、例えば、ブリッジ認証局ＣＡｂｒｉｄｇｅに対して公開鍵証明書を発行すると共に、ＣＡｂｒｉｄｇｅから公開鍵証明書を発行してもらうことにより、ＣＡｂｒｉｄｇｅとの間で相互認証を行っているものとする。このようにすることで、あるセキュリティドメインＳＤに属するＣＡと他のセキュリティドメインＳＤに属するＣＡとの間に、一方のＣＡが発行した公開鍵証明書の有効性を、他方のＣＡにて確認できるようにするためのパスが形成される。

【００３１】

次に、図１のＰＫＩシステムを構成する各装置について説明する。

【００３２】

まず、図３を用いて、ＥＥ装置（１１）を説明する。

【００３３】

ＥＥ装置（１１）は、処理部３０ａと記憶部３０ｂと、ＮＥＴ１６を介して他装置と通信を行うための通信部３０ｃと、ユーザ（ＥＥ）が作成した電子文書や他のＥＥから受け

取った電子文書の入出力やユーザからの指示の受付を行う入出力部３０ｄと、を有する。

【００３４】

処理部３０ａは、電子文書に対する署名を生成する署名生成部３１と、署名の検証を行う署名検証部３２と、ＥＥ装置の各部を統括的に制御する制御部３３と、を有する。

【００３５】

記憶部３０ｂは、利用者が作成した電子文書を保持する電子文書保持部３４と、秘密鍵（署名鍵）と、これと対になる公開鍵の公開鍵証明書と、当該ＥＥ装置を運用するＥＥが信頼するＣＡの自己署名証明書を保持する鍵保持部３５と、他のＥＥから受け取った署名付きの電子文書と公開鍵証明書を保持する検証対象保持部３６と、を有する。

【００３６】

このような構成において、制御部３３は、入出力部３０ｄを介してユーザから、電子文書保持部３４に保持してある電子文書を他のＥＥに送信すべき旨の指示を受け付けると、当該電子文書を電子文書保持部３４から読み出し、これを署名生成部３１に渡す。署名生成部３１は、鍵保持部３５に保持されている秘密鍵を用いて渡された当該電子文書に対する署名を生成する。

【００３７】

制御部３３は、電子文書保持部３４から読み出した電子文書に署名生成部３１で生成された署名を付して、署名付き電子文書を作成し、作成した署名付き電子文書と鍵保持部３５に保持されている公開鍵証明書とを、通信部３０ｃを介して、ユーザから指示された送信先のＥＥ装置へ送信する。制御部３３は、通信部３０ｃを介して、他のＥＥ装置から署名付き電子文書と公開鍵証明書を受け取ると、これらを関連づけて検証対象保持部３６に保持させると共に、これらの検証要求を署名検証部３２に通知する。

【００３８】

これを受けて、署名検証部３２は、検証対象保持部３６に保持されている署名付き電子文書の署名を、対応する公開鍵証明書を用いて検証する。署名検証部３２は、ＶＡ装置（１４）に、上記の署名検証に用いた公開鍵証明書の有効性の確認依頼を送信する。この際、必要に応じて、前記署名付き電子文書により行おうとしている電子手続に関するポリシー（例えば、取引額等の信頼度）の検証依頼を、前記確認依頼に含める。そして、ＶＡ装置（１４）において、上記ポリシーの検証も含めた当該公開鍵証明書の有効性が確認された場合にのみ、署名付き電子文書を正当なものとして扱い、必要に応じて入出力部３０ｄから出力する。

【００３９】

次に、図４を用いてＣＡ装置（１３）を説明する。

【００４０】

ＣＡ装置（１３）は、処理部４０ａと、記憶部４０ｂと、ＮＥＴ１６を介して他装置と通信を行うための通信部４０ｃと、公開鍵証明書等の入出力や当該装置の操作者からの指示の受付や処理結果の出力を行う入出力部３０ｄと、を有する。

【００４１】

処理部４０ａは、公開鍵証明書を発行する発行部４１と、発行部４１が発行した公開鍵証明書の管理を行う管理部４２と、ＣＡ装置の各部を統括的に制御する制御部４３と、を有する。

【００４２】

記憶部４０ｂは、発行部４１が発行した公開鍵証明書を保持する公開鍵証明書データベース４４と、公開鍵証明書データベース４４に保持されている各公開鍵証明書の発行先が記述されていた発行先管理リストを保持する発行先管理リスト保持部４５と、失効証明書リスト保持部４６と、を有する。

【００４３】

このような構成において、制御部４３は、入出力部４０ｄあるいは通信部４０ｃを介して公開鍵証明書の発行依頼を受け付けると、その旨を発行部４１に伝える。これを受けて、発行部４１は、これに対する公開鍵証明書を作成する。この際、ＣＡの秘密鍵で公開鍵

証明書に署名をする。また、必要に応じて、公開鍵証明書中に、当該公開鍵証明書の有効期限や、信頼しない他の認証局の名称（Ｎａｍｅ　Ｃｏｎｓｔｒａｉｎｓ）や、当該公開鍵証明書の有効性確認のために許容される最大パス長（パス上の許容最大認証局数）や、電子手続の取引額等を表したポリシーを記述する。そして、作成した公開鍵証明書を入出力部４０ｄあるいは通信部４０ｃを介して、郵送あるいは通信により、発行依頼元に渡す。また、この公開鍵証明書を公開鍵証明書データベース４４に登録すると共に、その発行先（つまり発行依頼元）の情報を、発行先管理リスト保持部４５に保持されている発行先管理リストに記述する。

【００４４】

制御部４３は、入出力部４０ｄあるいは通信部４０ｃを介して、公開鍵証明書の失効依頼を受け付けると、その旨を管理部４２に伝える。これを受けて、管理部４２は、失効対象の公開鍵証明書を公開鍵証明書データベース４３から削除すると共に、当該公開鍵証明書の発行先の情報を、発行先管理リスト保持部４４に保持されている発行先管理リストから削除する。そして、管理部４２は、失効依頼により公開鍵証明書データベース４３から削除した公開鍵証明書に関する情報が記述された失効証明書リストを、定期的に作成し、これを失効証明書リスト保持部４５に保持させる。なお、管理部４２は、作成した失効証明書リストに、次回の失効証明書リストの作成予定日時を記述するものとする。

【００４５】

また、制御部４３は、通信部４０ｃを介して、他装置より公開鍵証明書の失効情報の問い合わせを受け取ると、失効証明書リスト保持部４５に保持されている失効証明書リストを検索して、問い合わせのあった公開鍵証明書が失効しているか否かを調べる。そして、その結果を、通信部４０ｃを介して、問い合わせをした他装置に応答することもできる。（このような問い合わせと応答に用いる通信プロトコルとして、ＯＳＣＰ（Ｏｎｌｉｎｅ　Ｃｅｒｔｉｆｉｃａｔｉｏｎ　ｓｔａｔｕｓ　ｐｒｏｔｏｃｏｌ）がある）。

【００４６】

管理部４２は、公開鍵証明書データベース４４に格納されている各公開鍵証明書の有効期限を調査し、有効期限を過ぎている公開鍵証明書の発行先の情報を、発行管理リスト保持部４５に保持されている発行先管理リストから削除する処理も行う。

【００４７】

次に、図５を用いて、ＶＡ装置（１４）を説明する。

【００４８】

図示するように、ＶＡ装置（１４）は、処理部５０ａと、記憶部５０ｂと、ネットワークＮＥＴ１６を介して他装置と通信を行うための通信部５０ｃと、公開鍵証明書等の入出力やユーザよりの指示の受付けを行う入出力部５０ｄと、を有する。

【００４９】

処理部５０ａは、パス検索部５１と、パス検証部５２と、有効期限／失効状態調査部５３と、有効性確認部５４と、ＶＡ装置の各部を統括的に制御する制御部５５と、を有する。また、記憶部５０ｂは、パスデータベース５６と、失効証明書リスト作成予定日時データベース５７とを有する。このうちパスデータベース５６は、有効なパスデータベース５６Ａと、有効でないパスデータベース５６Ｂと、を有する。

【００５０】

パス検索部５１は、例えば定期的に、任意のＣＡをトラストアンカーＣＡとして、当該トラストアンカーＣＡから、ＥＥ証明書を発行する全てのＥＥ証明書発行ＣＡまでのパスを検索する。また、検出したパスのＥＥ証明書発行ＣＡが発行する、ＥＥ証明書に関する失効証明書リスト（ＣＲＬ）を取得する。トラストアンカーＣＡは、全てのＣＡ、または設定を設けることにより一部のＣＡとして、検索を行うことができる。

【００５１】

パス検証部５２は、パス検索部５１でパスの検索が行われる毎に、当該パス検索部５１で検出されたパスの検証を行う。また、パス検索部５１において当該パスに対応するＣＲＬを取得できた場合には、当該ＣＲＬの検証を行う。そして、当該パスの両端に位置する

こととなる、トラストアンカーＣＡの名称と、ＥＥ証明書発行ＣＡの名称とのペアに対応付けて、当該パスを構成する各ＣＡ名と、各証明書と、ＥＥ証明書に関するＣＲＬとを、当該検証結果に応じて、有効なパスデータベース５６Ａ、または有効でないデータベース５６Ｂに登録する。

【００５２】

有効期限／失効状態調査部５３は、有効なパスデータベース５６Ａに登録されているパス各々について、当該パスを構成する各公開鍵証明書の有効期限や失効の有無を調査する。そして、その結果に応じてパスデータベース５６を更新する。また、有効期限／失効状態調査部５３は、各ＣＡの失効証明書リスト保持部４６から入手した失効証明書リストに記述されている次回の失効証明書リスト作成予定日時を当該ＣＡに対応付けて、失効証明書リスト作成予定日時データベース５７に登録する。

【００５３】

有効性確認部５４は、ＥＥ装置からの依頼に従い、トラストアンカーＣＡを信頼の起点として、公開鍵証明書の有効性の確認を行う。

【００５４】

なお、図３～図５に示すＥＥ装置（１１）、ＣＡ装置（１３）とＶＡ装置（１４）の各々は、例えば、図６に示すような、ＣＰＵ６１と、メモリ６２と、ハードディスク等の外部記憶装置６３と、ＣＤ－ＲＯＭ等の可搬性を有する記憶媒体６９から情報を読み取る読み取り装置６４と、ＮＥＴ１６を介して他装置と通信を行うための通信装置６５と、キーボードやマウス等の入力装置６６と、モニタやプリンタ等の出力装置６７と、これらの各装置間のデータ送受を行うインタフェース６８とを備えた、一般的な電子計算機上に構築できる。

【００５５】

そして、ＣＰＵ６１が外部記憶装置６３からメモリ６２上にロードされた所定のプログラムを実行することにより、上述の各処理部を実現できる。すなわち、通信部３０ｃ、４０ｃ、５０ｃは、ＣＰＵ６１が通信装置６６を利用することにより、入出力部３０ｄ、４０ｄ、５０ｄは、ＣＰＵ６１が入力装置６６や出力装置６７や読み取り装置６４を利用することにより、そして、記憶部３０ｂ、４０ｂ、５０ｂは、ＣＰＵ６１がメモリ６２や外部記憶装置６３を利用することにより実現される。また、処理部３０ａ、４０ａ、５０ａは、ＣＰＵ６１のプロセスとして実現される。

【００５６】

上記所定のプログラムは、予め外部記憶装置６３に格納されていても良いし、上記電子計算機が利用可能な記憶媒体６９に格納されており、読み取り装置６４を介して、必要に応じて読み出され、あるいは、上記電子計算機が利用可能な通信媒体であるネットワークまたはネットワーク上を伝搬する搬送波を利用する通信装置６６と接続された他の装置から、必要に応じてダウンロードされて、外部記憶装置６３に導入されるものであってもよい。

【００５７】

次に、上記構成のＶＡ装置（１４）の動作を説明する。

【００５８】

本実施形態のＶＡ装置（１４）の動作は、パスの検索、検証および管理動作と、公開鍵証明書の有効性の確認動作とに分かれる。

【００５９】

図７～図８のフロー図を用いて、ＶＡ装置（１４）で行われるパスの検索、検証および管理動作を説明する。

【００６０】

制御部５５は、ＶＡの運営者によって定められた所定時間（例えば１日）を経過すると（ステップＳ１００１）、パスデータベース５６の登録内容を一旦クリアし（ステップＳ１００２）、パス検索部５１にパス検索を依頼する。これを受けて、パス検索部５１は、任意のＣＡをトラストアンカーＣＡとしたときの、ＥＥ証明書発行ＣＡまでのパスを検索

する（ステップＳ１００３）。

【００６１】

具体的には、パス検索部５１は、トラストアンカーＣＡにアクセスし、トラストアンカーＣＡが発行し、発行先管理リスト保持部４５に保持されている、公開鍵証明書の発行先の情報を入手する。そして、入手した各発行先がＣＡの場合は、各発行先にアクセスして、発行先管理リスト保持部４５に保持されている、各ＣＡが発行した公開鍵証明書の発行先をさらに調べる。この処理を、公開鍵証明書の発行先がＥＥとなるまで続けることにより、トラストアンカーＣＡからＥＥ証明書発行ＣＡまでのパスを検索する。ここで、パスのループにより上記の処理が無限に繰り返されるのを防止するため、あるＣＡから入手した発行先に、それまでに形成された部分パスに存在するＣＡが含まれる場合は、当該ＣＡを発行先とした上記の処理を行わないものとする。また、パスの端点に位置するＥＥへ証明書を発行したＣＡが発行するＣＲＬを取得する。

【００６２】

ステップＳ１００３でのパス検索処理を、各ＣＡが図２に示す関係にある場合を例に取り、より具体的に説明する。

【００６３】

まず、パス検索部５１は、トラストアンカーＣＡをＣＡｂｒｉｄｇｅとしてパス検索を行う。パス検索部５１は、ＣＡｂｒｉｄｇｅにアクセスし、発行先管理リスト保持部４５に保持されている、ＣＡｂｒｉｄｇｅが発行した公開鍵証明書の発行先の情報として、CA_{11}、CA_{21}の情報を入手する。

【００６４】

次に、パス検索部５１は、ＣＡｂｒｉｄｇｅから入手した発行先（CA_{11}、CA_{21}）のうちのいずれか１つに注目して、以下の処理を実行する。すなわち、注目した発行先がＣＡ（以下、注目ＣＡと呼ぶこととする）であるならば、ＣＡｂｒｉｄｇｅ－注目ＣＡという部分パスを設定する。そして、注目ＣＡの発行先管理リスト保持部４５にアクセスして、当該注目ＣＡが発行した公開鍵証明書の発行先の情報をさらに入手する。ここでは、注目した発行先がCA_{11}であるとして、ＣＡｂｒｉｄｇｅ－CA_{11}という部分パスを設定し、CA_{11}から、発行先の情報として、ＣＡｂｒｉｄｇｅ、CA_{12}、CA_{13}の情報を入手したものとする。

【００６５】

次に、パス検索部５１は、認証局CA_{11}から入手した発行先（ＣＡｂｒｉｄｇｅ、CA_{12}、CA_{13}）に、部分パス上のＣＡ（以下、ループＣＡと呼ぶこととする）が含まれているか否かを調べる。含まれている場合はその発行先を注目対象から除外する。したがって、ここでは、ＣＡｂｒｉｄｇｅを注目対象から除外することになる。次に、パス検索部５１は、CA_{11}から入手した発行先にＥＥが含まれるか否かを調べる。あるＣＡが発行した証明書の発行先にＥＥが含まれている場合、そのＣＡはＥＥ証明書発行ＣＡとなる。しかし、CA_{11}から入手した発行先にＥＥは含まれていないので、CA_{11}はＥＥ証明書発行ＣＡではない。したがって、ＣＡｂｒｉｄｇｅ－CA_{11}という部分パスを、ＥＥ証明書発行ＣＡまで延長すべく、CA_{11}から入手した、ループＣＡを除く発行先（CA_{12}、CA_{13}）のうちのいずれか１つに注目する。

【００６６】

注目した発行先がＣＡであるならば、それまでに設定した部分パスにこの注目ＣＡを接続した部分パスを設定する。そして、この注目ＣＡの発行先管理リスト保持部４５にアクセスして、当該注目ＣＡが発行した公開鍵証明書の発行先の情報をさらに入手する。ここでは、注目した発行先がCA_{12}であるとして、ＣＡｂｒｉｄｇｅ－CA_{11}－CA_{12}というパスを設定し、CA_{12}から、発行先の情報として、EE_1、EE_2を入手したものとする。

【００６７】

次に、パス検索部５１は、CA_{12}から入手した発行先（EE_1、EE_2）に、ループＣＡが含まれているか否かを調べる。含まれている場合はその発行先を注目対象から除外

する。ここでは、ループＣＡは含まれないので、パス検索部５１は、次の処理へ移行し、ＣＡ$_{12}$から入手した発行先にＥＥが含まれるか否かを調べる。ここで、入手した発行先はすべてＥＥであるので、ＣＡ$_{12}$はＥＥ証明書発行ＣＡである。そこで、このＣＡ$_{12}$を端点とするパスを、トラストアンカーＣＡｂｒｉｄｇｅから、ＥＥ証明書発行ＣＡ$_{12}$までのパス（ＣＡｂｒｉｄｇｅ－ＣＡ$_{11}$－ＣＡ$_{12}$）として検出する。

【００６８】

さらに、パス検索部５１は、ＥＥ証明書発行ＣＡまでのパスを検出した場合、当該ＥＥ証明書発行ＣＡ$_{12}$が発行するＣＲＬを、失効証明書リスト保持部４６にアクセスし、取得する。

【００６９】

次に、パス検索部５１は、検出したパス上の端点に位置するＣＡ$_{12}$から入手した発行先の情報の中に、未だ注目していない発行先（ループＣＡ以外のＣＡ）があるか否かを調べ、そのような発行先があれば、これを注目ＣＡとして、上記の処理を続ける。一方、そのような発行先がなければ、１つ前に位置するＣＡ$_{11}$から入手した発行先の情報の中に、未だ注目していない発行先（ループＣＡ以外のＣＡ）があるか否かを調べる。そして、そのような発行先があれば、これを注目ＣＡとして、上記の処理を続ける。ここでは、ＣＡ$_{11}$から入手した発行先の情報のうち、ＣＡ$_{13}$について未だ注目していないので、これを注目ＣＡとして上記の処理を行うことにより、ＣＡｂｒｉｄｇｅからＥＥ証明書発行ＣＡ$_{13}$までのパス（ＣＡｂｒｉｄｇｅ－ＣＡ$_{11}$－ＣＡ$_{13}$）およびＣＡ$_{13}$が発行するＣＲＬを検出する。

【００７０】

このように、パス検索部５１は、上記の処理を、検出したパス上に位置する全てのＣＡ各々について、当該ＣＡから入手した発行先の情報の中に、未だ注目していない発行先（ループＣＡ以外のＣＡ）がなくなるまで続けることにより、ＣＡｂｒｉｄｇｅから各ＥＥ証明書発行ＣＡまでのパスを検出する。

【００７１】

以上が、任意のＣＡをトラストアンカーＣＡとした場合のステップ１００３の処理である。

【００７２】

後述するように、ＣＡ$_{11}$、ＣＡ$_{21}$各々のＣＡをトラストアンカーＣＡとした場合についても、同様にパス検索を行う。

【００７３】

制御部５５は、パス検索部５１によりパスが検出されると（ステップＳ１００４でＹｅｓ）、パス検証部５２にパスの検証を依頼する。これを受けて、パス検証部５２は、パス検索部５１により検出されたパスの検証を行う（ステップＳ１００５）。

【００７４】

具体的には、パス検索部５１により検出されたパス各々について、以下の処理を行う。

【００７５】

すなわち、まず、パス検証部５２は、パス上の各ＣＡの公開鍵証明書データベース４４にアクセスし、各ＣＡが当該パス上の１つ次に位置するＣＡ（アクセス先ＣＡがＥＥ証明書発行ＣＡの場合はＥＥ）に対して発行した公開鍵証明書を入手する。

【００７６】

次に、パス検証部５２は、パスの最後に位置するＥＥ証明書発行ＣＡが発行した公開鍵証明書の署名を、ＥＥ証明書発行ＣＡの公開鍵証明書で検証し、検証できた場合は当該ＥＥ証明書発行ＣＡの公開鍵証明書の署名をさらに１つ前に位置するＣＡの公開鍵証明書で検証する。この処理を、当該１つ前に位置する認証局ＣＡがトラストアンカーＣＡとなるまで続けることにより、当該パスを検証する。さらに、当該ＥＥ証明書発行ＣＡが発行するＣＲＬについて、当該ＥＥ証明書発行ＣＡの公開鍵証明書で検証する。

【００７７】

例えば、図２においてＣＡｂｒｉｄｇｅからＥＥ証明書発行ＣＡ$_{12}$までのパス（ＣＡ

ｂｒｉｄｇｅ－ＣＡ１１－ＣＡ１２）およびＣＲＬを検証する場合、まず、ＥＥ証明書発行ＣＡ１２の公開鍵証明書の署名を、パス中でＣＡ１２の１つ前に位置するＣＡ１１の公開鍵証明書を用いて検証する。そして、検証できた場合は、ＣＡ１１の公開鍵証明書の署名を、パス中でＣＡ１１より１つ前に位置するＣＡｂｒｉｄｇｅの公開鍵証明書を用いて検証する。そして、この検証できた場合、さらに、当該ＥＥ証明書発行ＣＡ１２が発行するＣＲＬについて、当該ＥＥ証明書発行ＣＡ１２の公開鍵証明書で検証する。この、パスおよびＣＲＬの検証できた場合に、ＣＡｂｒｉｄｇｅからＥＥ証明書発行ＣＡ１２までのパスが仮に検証できたものとする。

【００７８】

次に、パス検証部５２は、パスが仮に検証できたならば、当該パス上の各認証局ＣＡから入手した公開鍵証明書中に、信頼しない他の認証局の名称（Ｎａｍｅ　Ｃｏｎｓｔｒａｉｎｓ）や当該公開鍵証明書の有効性確認のために許容される最大パス長（パス上の許容最大認証局数）などの制限の記述があるか否かを調べる。そのような記述がある場合は、当該パスがその制限を満たしているか否かを調べ、満たしている場合にのみ、当該パスが検証できたものとする。

【００７９】

さて、制御部５５は、上記のようにして、パス検索部５１により検出されたパス各々に対するパス検証部５２での検証が終了したならば、登録処理を行う。制御部５５は、パス検証部５２でパスが検証できた場合（ステップＳ１００６でＹｅｓ）、当該パスをトラストアンカーＣＡと、ＥＥ証明書発行ＣＡと、ＥＥ証明書発行ＣＡが発行するＣＲＬと、に対応付けて、有効なパスデータベース５６Ａに登録し（ステップＳ１００７）、ステップＳ１００３へ移行する。また、制御部５５は、パス検証部５２でパスが検証できなかった場合（ステップＳ１００６でＮｏ）、当該パスをトラストアンカーＣＡと、ＥＥ証明書発行ＣＡと、ＥＥ証明書発行ＣＡが発行するＣＲＬと、に対応付けて、有効でないパスデータベース５６Ｂに登録し（ステップＳ１００８）、ステップ１００３へ移行する。

【００８０】

制御部５５は、ステップＳ１００３～ステップＳ１００８のステップを、パスが検出されなくなる（ステップＳ１００４でＮｏ）まで繰り返し、パスデータベース５６を作成する。このとき、全てのＣＡをトラストアンカーとして、対応する全てのパスを検索する。ＣＡの構成が図２の場合は、ＣＡ１１、ＣＡ２１、ＣＡｂｒｉｄｇｅの３つのＣＡをトラストアンカーとして、各々に対応する全てのパスを検索する。

【００８１】

ステップＳ１００３～ステップＳ１００８までの処理を行った結果、各ＣＡが図２に示す関係にある場合、パス検索部５１により検出される、全てのパスは、図９に示すとおりとなる。

【００８２】

一方、有効期限／失効状態調査部５３は、有効なパスデータベース５６Ａに登録されている公開鍵証明書の中に、有効期限切れの公開鍵証明書があるか否かを調べる（ステップＳ１００９）。有効期限切れの公開鍵証明書がある場合は、当該公開鍵証明書の発行元ＣＡの公開鍵証明書データベース４４にアクセスして、当該公開鍵証明書の発行先に対して新たに発行された公開鍵証明書を検索する（ステップＳ１０１０）。

【００８３】

そして、そのような公開鍵証明書が前記発行元ＣＡの公開鍵証明書データベース４４中になければ、前記有効期限切れの公開鍵証明書に対応付けて登録されているパスに関する情報を、有効なパスデータベース５６Ａから削除し、有効でないパスデータベース５６Ｂへ登録する（ステップＳ１０１１）。一方、そのような公開鍵証明書が前記発行元ＣＡの公開鍵証明書データベース４４中にあればこれを入手する。そして、前記有効期限切れの公開鍵証明書に対応付けて、有効なパスデータベース５６Ａに登録されているパスの検証を、当該有効期限切れの公開鍵証明書の代わりに新たに入手した公開鍵証明書を用いて、上記のステップＳ１００５と同じ要領で行う（ステップＳ１０１２）。

【００８４】
さて、パスが検証できた場合（ステップＳ１０１３でＹｅｓ）は、当該パスに対応付けられて有効なパスデータベース５６Ａに登録されている前記有効期限切れの公開鍵証明書を、新たに入手した公開鍵証明書に置き換える（ステップＳ１０１４）。一方、パスが検証できなかった場合（ステップＳ１０１３でＮｏ）は、前記有効期限切れの公開鍵証明書に対応付けて登録されているパスを、有効なパスデータベース３１から削除し、新たに入手した公開鍵証明書に置き換えたパスを、有効でないパスデータベースに置き換える（ステップＳ１０１５）。
【００８５】
次に、有効期限／失効状態調査部５３は、失効証明書リスト作成予定日時データベース５７を調べ、既に経過した失効証明書リスト作成予定日時に対応付けられているＣＡを検索する（ステップＳ１０１６）。そのような認証局ＣＡが存在する場合（ステップＳ１０１７でＹｅｓ）は、当該ＣＡの失効証明書リスト保持部４６にアクセスして、当該ＣＡが発行した最新の失効証明書リストを入手する（ステップＳ１０１８）。そして、失効証明書リスト作成予定日時データベース５７にて、当該認証局ＣＡに対応付けて登録されている失効証明書リスト作成予定日時を、入手した最新の失効証明書リストに記述されている失効証明書リスト作成予定日時に更新する（ステップＳ１０１９）。
【００８６】
それから、有効期限／失効状態調査部５３は、入手した最新の失効証明書リストに記述されている公開鍵証明書が、有効なパスデータベース５６Ａに登録されているか否かを調べ（ステップＳ１０２０）、登録されている場合は、当該公開鍵証明書に対応付けられているパスに関する情報を有効なパスデータベース５６Ａから削除し、有効でないパスデータベース５６Ｂへ登録する（ステップＳ１０２１）。
【００８７】
次に、公開鍵証明書の有効性の確認動作について説明する。
【００８８】
図１０～図１１は、本実施形態のＶＡ装置（１４）で行われる公開鍵証明書の有効性の確認動作を説明するためのフロー図である。
【００８９】
制御部５５は、通信部５０ｃを介して、ＥＥから、少なくとも当該ＥＥが信頼するトラストアンカーＣＡの名称を含んだ、他ＥＥの公開鍵証明書の有効性確認依頼を受け取ると（ステップＳ２００１）、その旨を有効性確認部５４に通知する。
【００９０】
これを受けて、有効性確認部５４は、証明書の有効性確認依頼の記述から特定される、トラストアンカーＣＡと、当該証明書を発行したＥＥ証明書発行ＣＡと、に対応付けられたパスが、
有効なパスデータベース５６Ａに登録されているか否かを調べる（ステップＳ２００２）。
【００９１】
その結果、証明書の有効性確認依頼に記述された、トラストアンカーＣＡと、当該証明書を発行したＥＥ証明書発行ＣＡと、に対応付けられたパスが、有効なパスデータベース５６Ａに登録されていることが確認できたならば（ステップＳ２００２でＹｅｓ）、有効性確認部５４は、当該パスの端点であるＥＥ証明書発行ＣＡの公開鍵証明書を用いて、ＥＥ証明書の署名検証を行う。さらに、有効性確認部５４は、当該パスと対応付けられて登録されているＣＲＬを用いて、ＥＥ証明書が失効していないかどうかを確認する（ステップＳ２００３）。
【００９２】
ＥＥ証明書の署名検証が失敗した場合（ステップ２００３でＮｏ）、又はＥＥ証明書がＣＲＬに記述され失効されていた場合、有効性確認部５４は、ＥＥ証明書有効でないものと判断し、その旨を、通信部５０ｃを介して、依頼元のＥＥに通知する（ステップＳ２０

０９）。

【００９３】

一方、各証明書には、信頼しない認証局名による制限や、当該公開鍵証明書の有効性確認のために許容される最大パス長（パス上の許容最大認証局数）を記述できる拡張項目がある。ステップ２００３で、ＥＥ証明書の署名検証および、失効していないかどうかの確認が成功した場合（Ｙｅｓの場合）、ＥＥ証明書および当該パスに含まれる各ＣＡの証明書に、上述の制限が記述されているか否かをさらに調べる（ステップＳ２００４）。

【００９４】

そのような制限の記述がない場合、ステップＳ２００６に移行する。

【００９５】

そのような制限の記述がある場合は、ステップＳ２００５に移行して、ＥＥ証明書がその制限に反していないか否かを調べる。ＥＥ証明書が制限事項に反している場合、有効性確認部５４は、公開鍵証明書が有効性でない旨を、通信部５０ｃを介して、依頼元のＥＥ装置（１１）に通知する（ステップＳ２００９）。ＥＥ証明書が制限事項に反していない場合は、ステップＳ２００６に移行する。

【００９６】

ステップＳ２００６では、有効性確認部５４は、ＥＥ装置（１１）から受け取った確認依頼に、当該ＥＥが行おうとしている電子手続の取引額等を示したポリシーが含まれているか否かを調べる。

【００９７】

ポリシーが含まれている場合は、ＥＥ証明書および当該パスを構成する各公開鍵証明書中に、前記ポリシーを満たすポリシーの記述があるか否かをさらに調べる（ステップＳ２００７）。

【００９８】

ＥＥ証明書および当該パスに、前記ポリシーを満たすポリシーの記述がない場合は、ＥＥ証明書を、依頼元のＥＥが行おうとしている電子手続のための公開鍵証明書の有効性確認に利用できないものと判断し、公開鍵証明書が有効でない旨を、通信部５０ｃを介して、依頼元のＥＥ装置（１１）に通知する（ステップＳ２００９）。

【００９９】

一方、ＥＥから受け取った確認依頼に当該ＥＥが行おうとしている電子手続を示すポリシーが含まれていない場合（ステップＳ２００６でＮｏ）、あるいは、含まれていても、当該パスおよびＥＥ証明書に記述されているポリシーが前記ポリシーを満たす場合（ステップＳ２００７でＹｅｓ）は、公開鍵証明書は有効であると判断し、公開鍵証明書が有効である旨を、通信部５０ｃを介して、依頼元のＥＥに通知する（ステップＳ２００８）。

【０１００】

また、ステップＳ２００２において、証明書の有効性確認依頼に記述された、トラストアンカーＣＡと、当該証明書を発行したＥＥ証明書発行ＣＡと、に対応付けられたパスが、有効なパスデータベース５６Ａに登録されていない場合（ステップＳ２００２でＮｏ）、当該パスが有効でないパスデータベース５６Ｂに登録されているか否かを確認する（ステップＳ２０１０）。有効でないデータベース５６Ｂに、当該パスが登録されていない場合（ステップＳ２０１０でＮｏ）、図１１のステップＳ２０１２に移行する。

【０１０１】

ステップＳ２０１２では、パス検索部５１が、確認依頼記述されたトラストアンカーＣＡから、確認対象のＥＥ証明書までのパスを検索する。この検索は、パス検索部５１が予め定められた所定の規則に従って行うものとは異なり、臨時に行う。

【０１０２】

パス検索部５１が、トラストアンカーＣＡから、ＥＥ証明書までのパスを検出しなかった場合（ステップＳ２０１３でＮｏ）、有効性確認部５４は、ＥＥ証明書が有効性でない旨を、通信部５０ｃを介して、依頼元のＥＥに通知する（ステップＳ２０１９）。一方、パス検索部５１が、トラストアンカーＣＡから、ＥＥ証明書までのパスを検出した場合（

ステップＳ２０１３でＹｅｓ）、パス検証部５２において、検出したパスを検証する（ステップＳ２０１４）。

【０１０３】

検出したパスが検証できた場合（ステップＳ２０１５でＹｅｓ）、当該パスのトラストアンカーＣＡから、ＥＥ証明書発行ＣＡまでのパスと、ＥＥ証明書発行ＣＡが発行するＣＲＬを、有効なパスデータベースへ登録する（ステップＳ２０１６）。そして、有効性確認部５４は、ＥＥ証明書が有効性である旨を、通信部５０ｃを介して、依頼元のＥＥに通知する（ステップＳ２０１７）。

【０１０４】

一方、ステップ２０１５において、検出したパスが検証できなかった場合（Ｓ２０１５でＮｏ）、当該パスのトラストアンカーＣＡから、ＥＥ証明書発行ＣＡまでのパスと、ＥＥ証明書発行ＣＡが発行するＣＲＬを、有効でないパスデータベースへ登録する（ステップＳ２０１８）。そして、ステップＳ２０１１へ移行し、これまで検出したパス以外に、パスが存在するかどうか検索を行い、それ以降のステップを同じように実行する。

【０１０５】

また、図１０のステップ２０１０において、有効でないパスデータベース５６Ｂに、当該パスが登録されていた場合（ステップＳ２０１０でＹｅｓ）、さらに当該登録されていたパス以外に、有効性確認依頼に対応するパスを検出した場合（ステップＳ２０１１でＹｅｓ）、ステップＳ２０１４に移行し、当該検出したパスの検証、登録処理を行う（ステップＳ２０１４～ステップＳ２０１９）。

【０１０６】

一方、当該登録されていたパス以外に、有効性確認依頼に対応するパスを検出しなかった場合（ステップＳ２０１１でＮｏ）、有効性確認部５４は、公開鍵証明書が有効性でない旨を、通信部５０ｃを介して、依頼元のＥＥに通知する（ステップＳ２００９）。

【０１０７】

以上の実施形態では、トラストアンカーＣＡから各ＥＥ証明書発行ＣＡまでのパスの検索および検証を、ＥＥからの公開鍵証明書の有効性確認依頼とは独立した所定の規則に従って、例えば定期的に行う。

【０１０８】

検索、検証したパスは、有効なパスまたは有効でないパスに分類し、パスデータベースに登録しておく。そして、有効性確認依頼に対応するパスが、有効なパスとして登録されているか、有効でないパスとして登録されているかを調べることにより、当該ＥＥ証明書が有効であるか否かを判断する。

【０１０９】

このとき、当該有効性確認依頼に対するパスが、有効でないパスに登録されている場合は、当該有効でないパス以外に対応するパスがあるかどうかを、検索、検証し、確認を行う。また、当該有効性確認依頼に対するパスが、パスデータベースに登録されていない場合には、パス検索、検証を臨時に行う。したがって、認証局の構成が変化した場合にも、最新のパス情報により検証を行うため、適切な有効性確認結果を応答することができる。また、有効でないパスをキャッシュしておくことにより、公開鍵証明書を受け付けてから、当該有効性を確認するまでにかかる時間を短縮することができる。

【０１１０】

また、本実施形態では、パスを登録する際に、パスの端点となるＥＥ証明書発行ＣＡが発行するＣＲＬを、当該ＣＲＬの検証結果とともに登録しておく。そして、あるＥＥから公開鍵証明書の有効性確認依頼を受けた場合、当該ＥＥ証明書が失効しているかどうかを、当該ＣＲＬを用いて確認する。したがって、公開鍵証明書の有効性確認にかかる時間をより短縮することができる。

【０１１１】

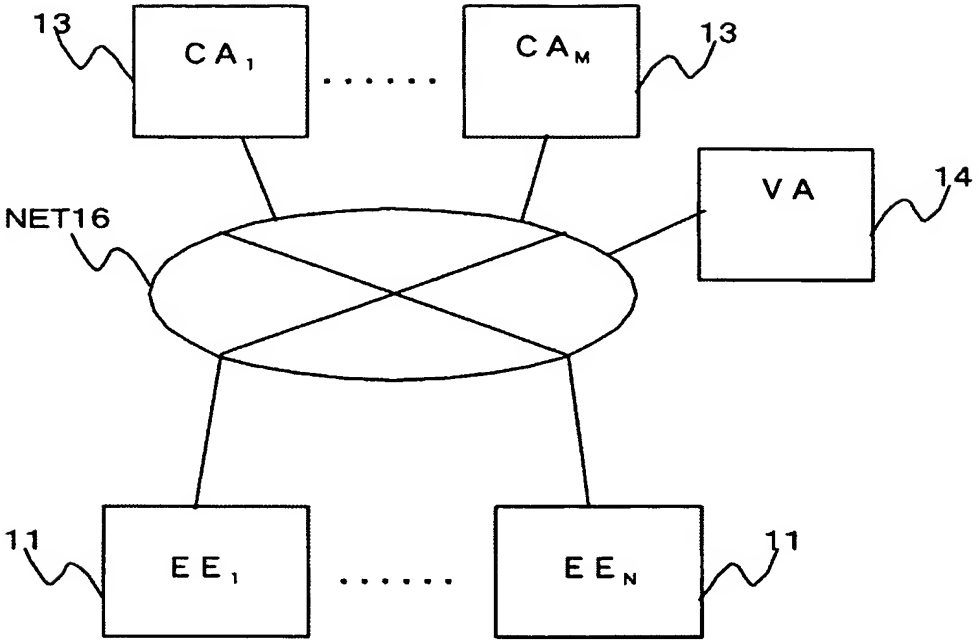
なお、本発明は、上記の実施形態に限定されるものではなく、その要旨の範囲内で数々の変形が可能である。

【０１１２】
例えば、上記の実施形態では、ＶＡは、パスをデータベースに登録する際に、有効なパスデータベース５６Ａと、有効でないパスデータベース５６Ｂの、二つのデータベースに分類して登録している。しかし、一つのデータベースに、当該パスの状態を示すフラグをつけて、有効なパスと、有効でないパスを分類してもよい。
【０１１３】
また、上記の実施形態では、説明をわかりやすくするため、図２に示すように、ＥＥ証明書発行ＣＡはＥＥに対してのみ公開鍵証明書を発行し、その他のＣＡはＣＡに対してのみ公開鍵証明書を発行するものと仮定したが、ＥＥとＣＡの両方に対して公開鍵証明書を発行するＣＡを含む場合でも、本発明は適用できる。
【０１１４】
また、上記の実施形態では、説明を分かりやすくするため、図２に示すように、ＣＡの構成を階層構造と仮定したが、ＣＡの構成がより複雑なメッシュ構造となっている場合でも、本発明は適用できる。
【図面の簡単な説明】
【０１１５】
【図１】本発明の一実施形態が適用されたＰＫＩシステムの概略構成を示す図である。
【図２】図１に示すＰＫＩシステムでの各ＣＡの関係の一例を示す図である。
【図３】図１に示すＥＥの概略構成を示す図である。
【図４】図１に示すＶＡの概略構成を示す図である。
【図５】図１に示すＶＡの概略構成を示す図である。
【図６】図３～図５に示すＥＥ、ＣＡおよびＶＡの各々のハードウェア構成例を示す図である。
【図７】図５に示すＶＡで行われるパスの検索、検証および管理動作を説明するためのフロー図である。
【図８】図５に示すＶＡで行われるパスの検索、検証および管理動作を説明するためのフロー図である。
【図９】各ＣＡが図２に示す関係にある場合に、ＶＡのパス検索部５１で検出される全てのパスを示す図である。
【図１０】図５に示すＶＡで行われる公開鍵証明書の有効性の確認動作を説明するためのフロー図である。
【図１１】図５に示すＶＡで行われる公開鍵証明書の有効性の確認動作を説明するためのフロー図である。
【図１２】従来のＰＫＩにおいて、認証局が複数ある場合におけるＣＡの関係の一例を示す図である。
【符号の説明】
【０１１６】
ＣＡ・・・認証局，ＶＡ・・・証明書の有効性確認局，ＥＥ・・・エンドエンティティ，ＳＤ・・・セキュリティドメイン，１１・・・EE_1，１２・・・EE_N，１３・・・CA_1，１４・・・CA_M，１５・・・ＶＡ，１６・・・ＶＡ，３０ａ，４０ａ，５０ａ・・・処理部，３０ｂ，４０ｂ，５０ｂ・・・記憶部，３０ｃ，４０ｃ，５０ｃ・・・通信部，３０ｄ，４０ｄ，５０ｄ・・・入出力部，３１・・・署名生成部，３２・・・署名検証部，３３，４３，５５・・・制御部，３４・・・電子文書保持部，３５・・・鍵保持部，３６・・・検証対象保持部，４１・・・発行部，４２・・・管理部，４４・・・公開鍵証明書データベース，４５・・・発行先管理リスト保持部，４６・・・失効証明書リスト保持部，５１・・・パス検索部，５２・・・パス検証部，５３・・・有効期限／執行状態調査部，５４・・・有効性確認部，５６・・・パスデータベース，５６Ａ・・・有効なパスデータベース，５６Ｂ・・・有効でないパスデータベース，５７・・・失効証明書リスト作成予定日時データベース，６１・・・ＣＰＵ、６２・・・メモリ、６３・・・外部記

憶装置、６４・・・読取装置、６５・・・通信装置、６６・・・入力装置、６７・・・出力装置、６８・・・インタフェース、６９・・・記憶媒体。

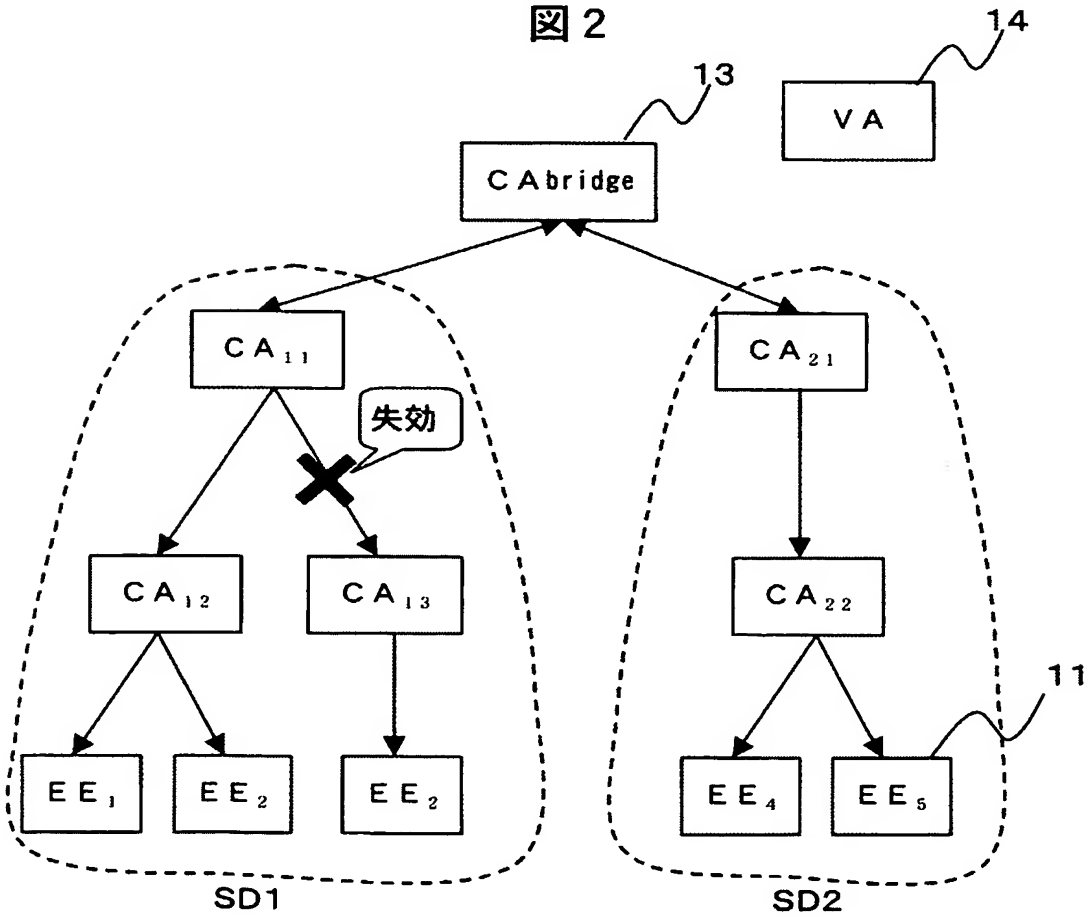
【書類名】図面
【図１】

図１



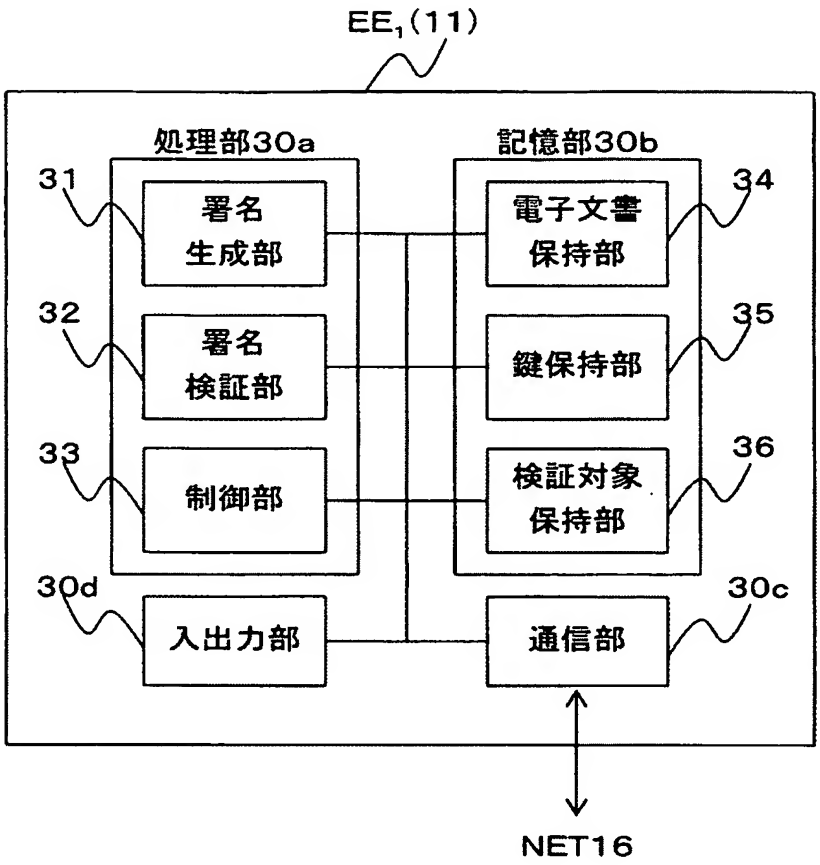
【図２】

図２



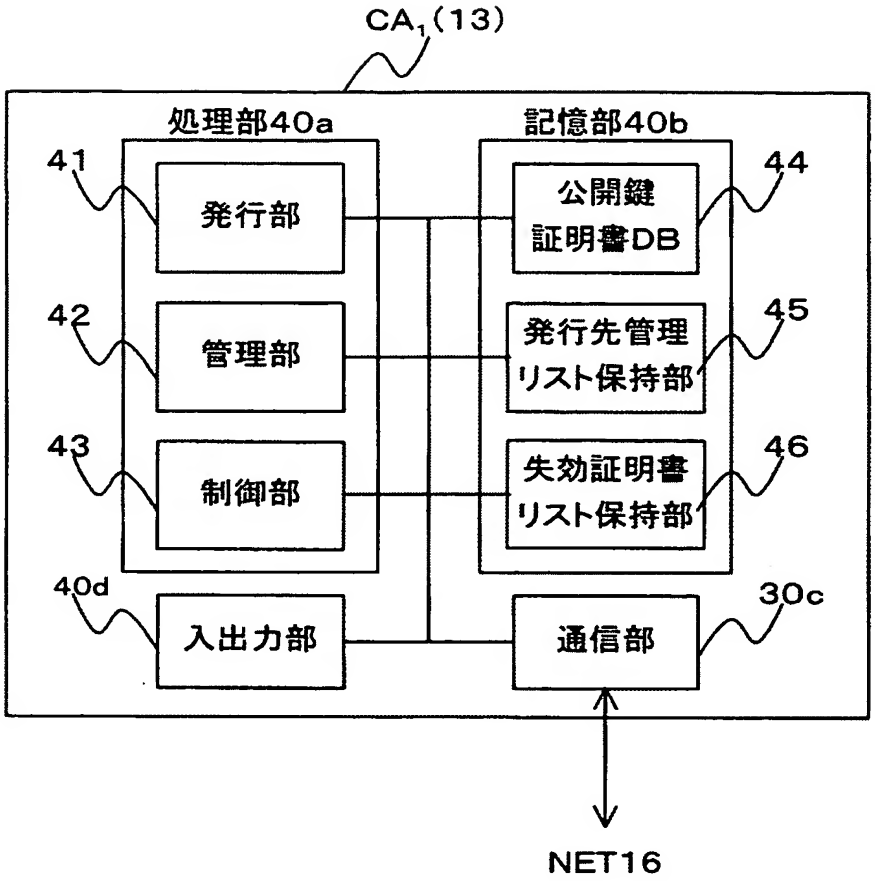
【図３】

図３


EE1(11)
処理部30a
記憶部30b
31
署名
生成部
32
署名
検証部
33
制御部
30d
入出力部
34
電子文書
保持部
35
鍵保持部
36
検証対象
保持部
30c
通信部
NET16

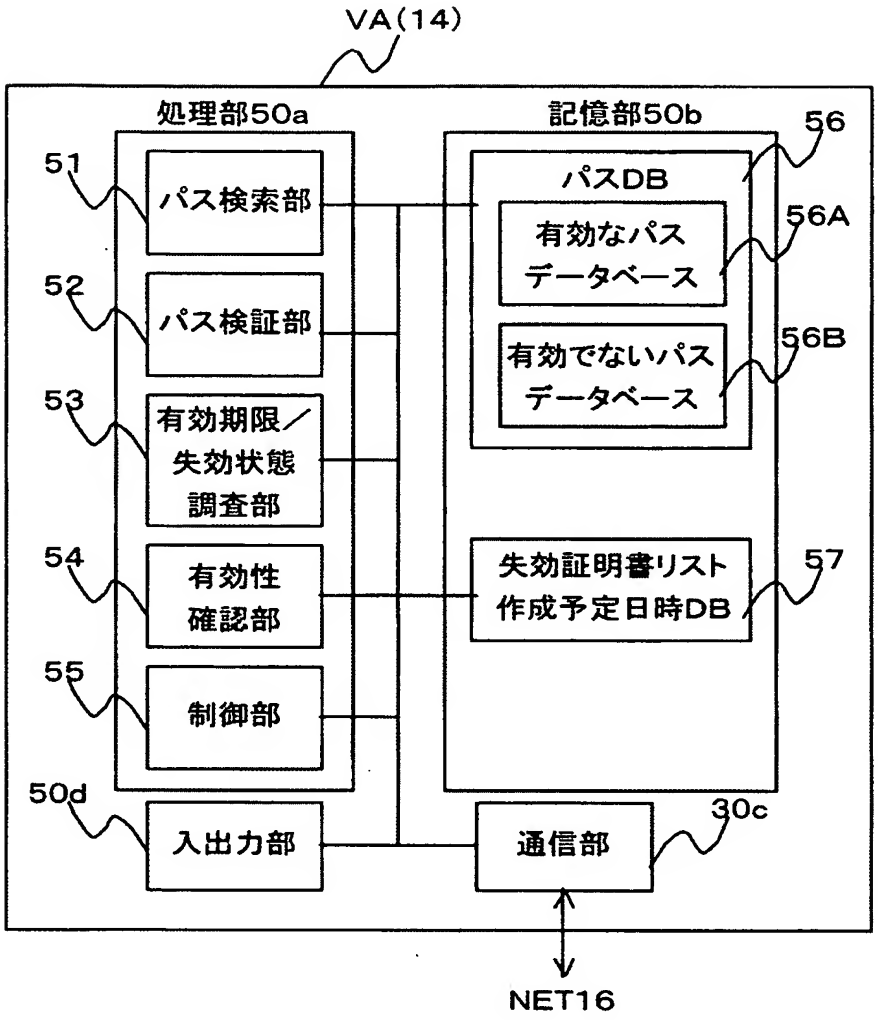
【図４】

図４


CA1(13)
処理部40a
記憶部40b
41
発行部
42
管理部
43
制御部
40d
入出力部
公開鍵
証明書DB
44
発行先管理
リスト保持部
45
失効証明書
リスト保持部
46
通信部
30c
NET16

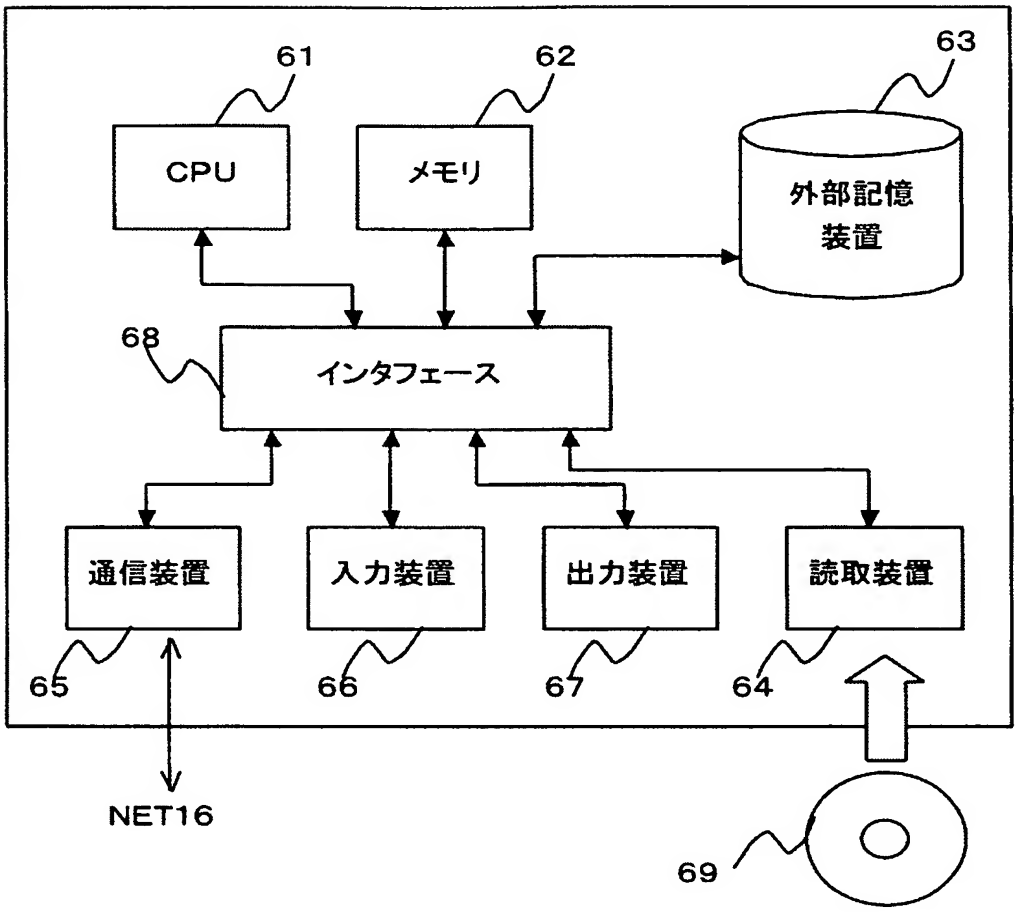
【図５】

図５


VA(14)
処理部50a
記憶部50b
56
51
パス検索部
パスDB
有効なパス
データベース
56A
52
パス検証部
有効でないパス
データベース
56B
53
有効期限／
失効状態
調査部
54
有効性
確認部
失効証明書リスト
作成予定日時DB
57
55
制御部
50d
入出力部
通信部
30c
NET16

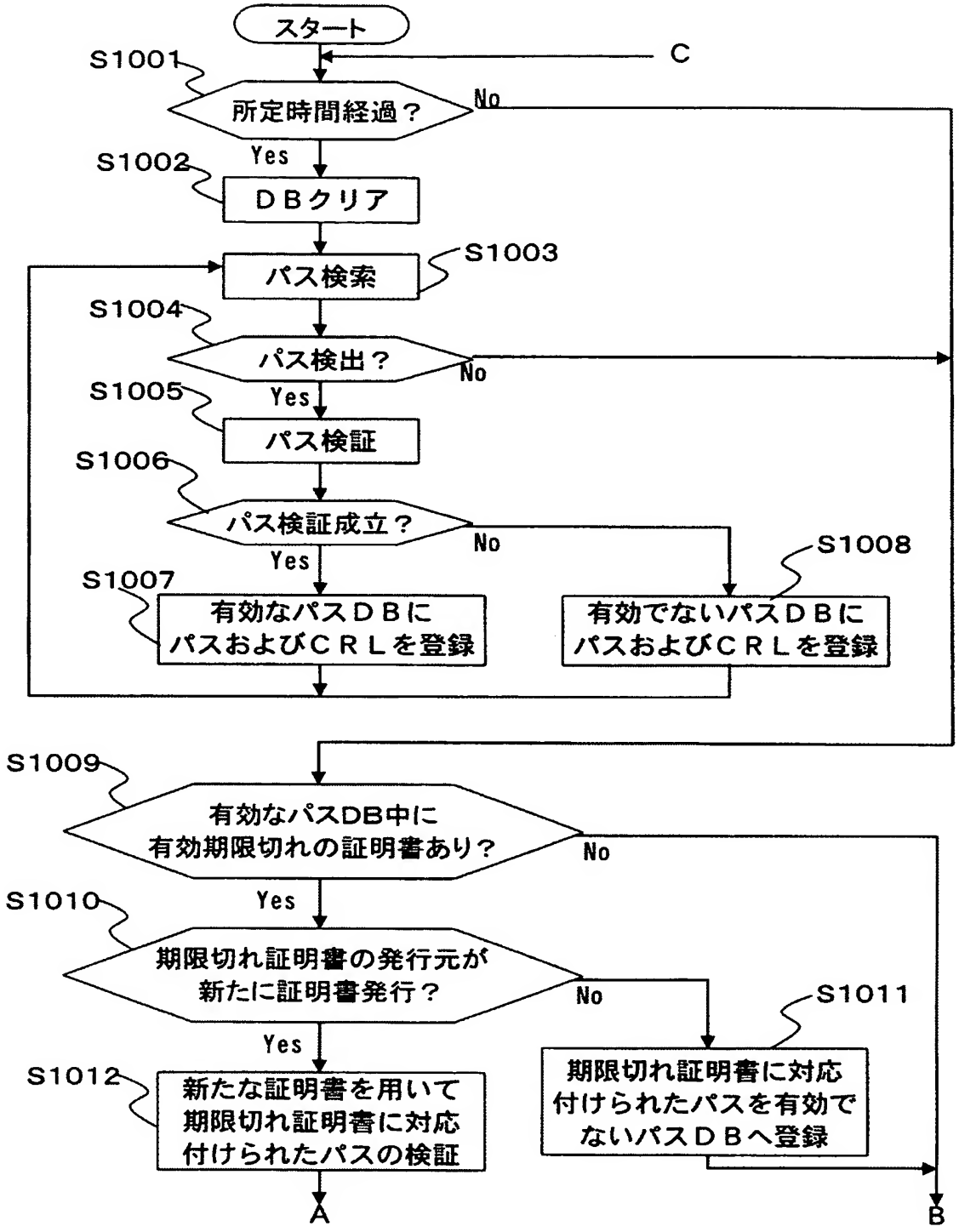
【図６】

図６


61
62
63
CPU
メモリ
外部記憶
装置
68
インタフェース
通信装置
入力装置
出力装置
読取装置
65
66
67
64
NET16
69

【図７】

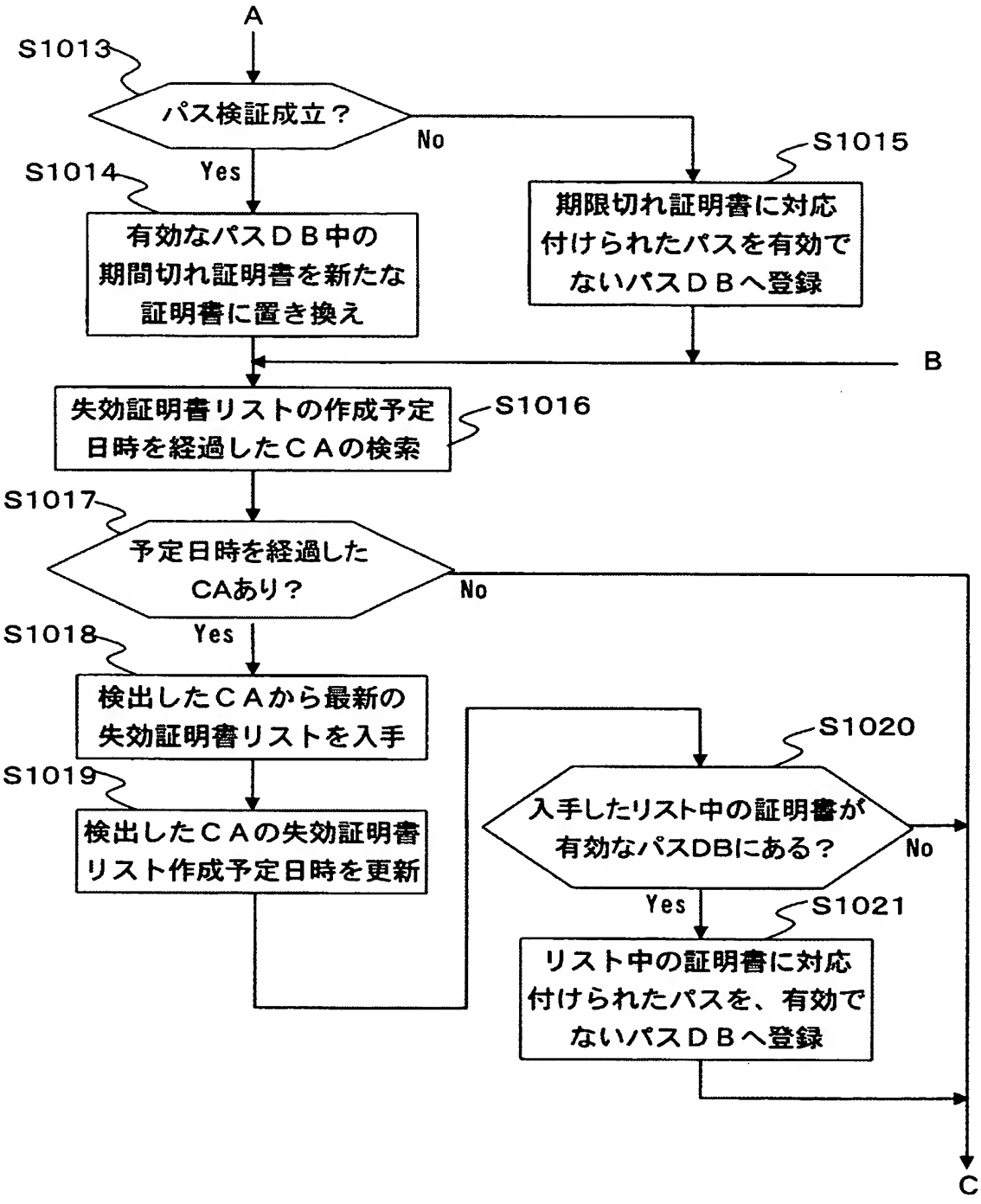
図７　　パスの検索、検証、および管理動作



【図８】

図８

パスの検索、検証、および管理動作



【図９】

図９

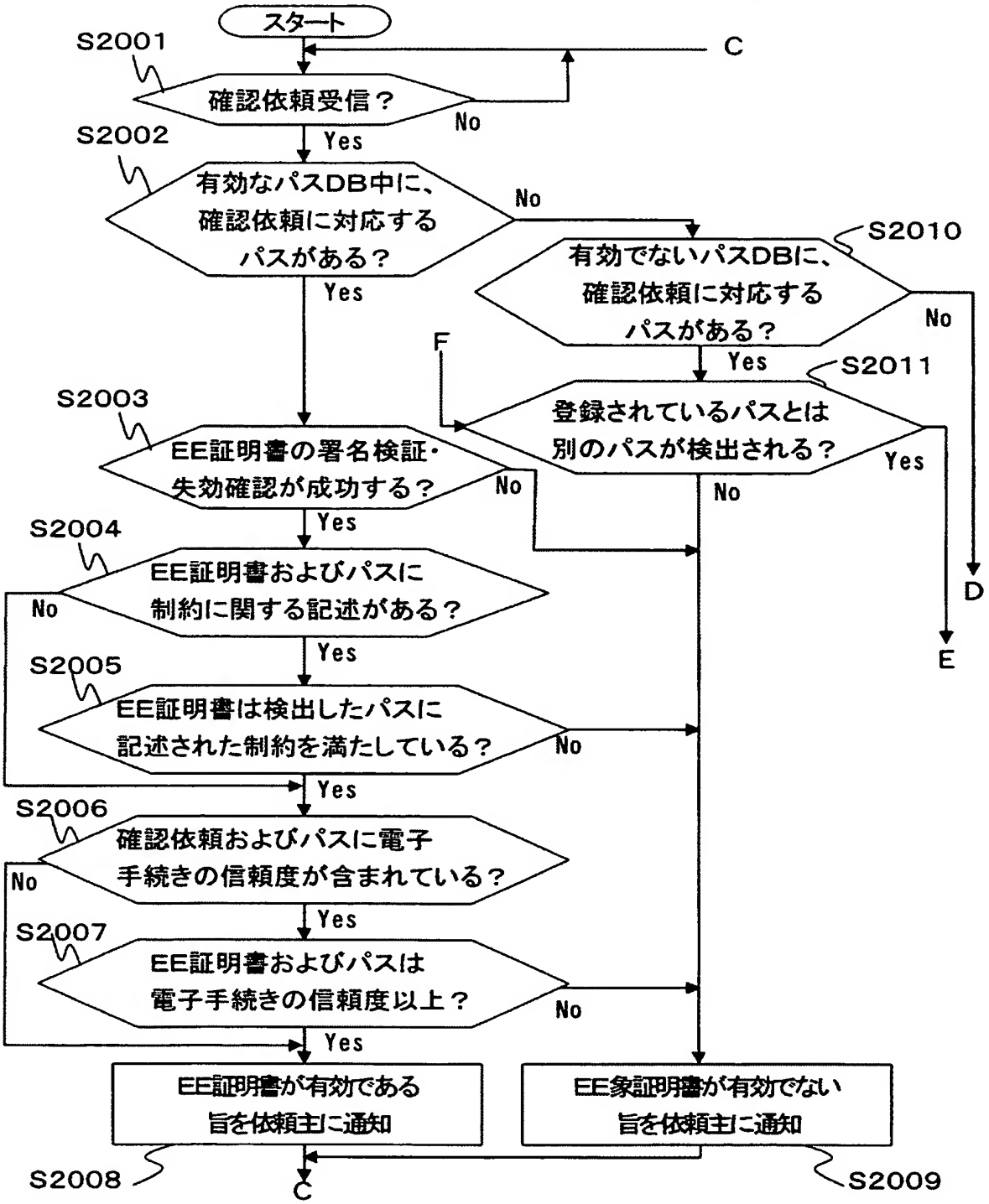
パス検索結果（図2の場合）

有効なパス			
トラストアンカーＣＡ	ＥＥ証明書発行ＣＡ	パス	ＣＲＬ
CA_{11}	CA_{12}	CA_{11}－CA_{12}	有効
	CA_{22}	CA_{11}－CAbridge－CA_{21}－CA_{22}	有効
CA_{21}	CA_{12}	CA_{21}－CAbridge－CA_{11}－CA_{12}	有効
	CA_{22}	CA_{21}－CA_{22}	有効
CAbridge	CA_{12}	CAbridge－CA_{11}－CA_{12}	有効
	CA_{22}	CAbridge－CA_{21}－CA_{22}	有効

有効でないパス			
トラストアンカーＣＡ	ＥＥ証明書発行ＣＡ	パス	ＣＲＬ
CA_{11}	CA_{13}	CA_{11}－CA_{12}	無効
CA_{21}	CA_{12}	CA_{21}－CAbridge－CA_{11}－CA_{12}	無効
CAbridge	CA_{12}	CAbridge－CA_{11}－CA_{12}	無効

【図１０】

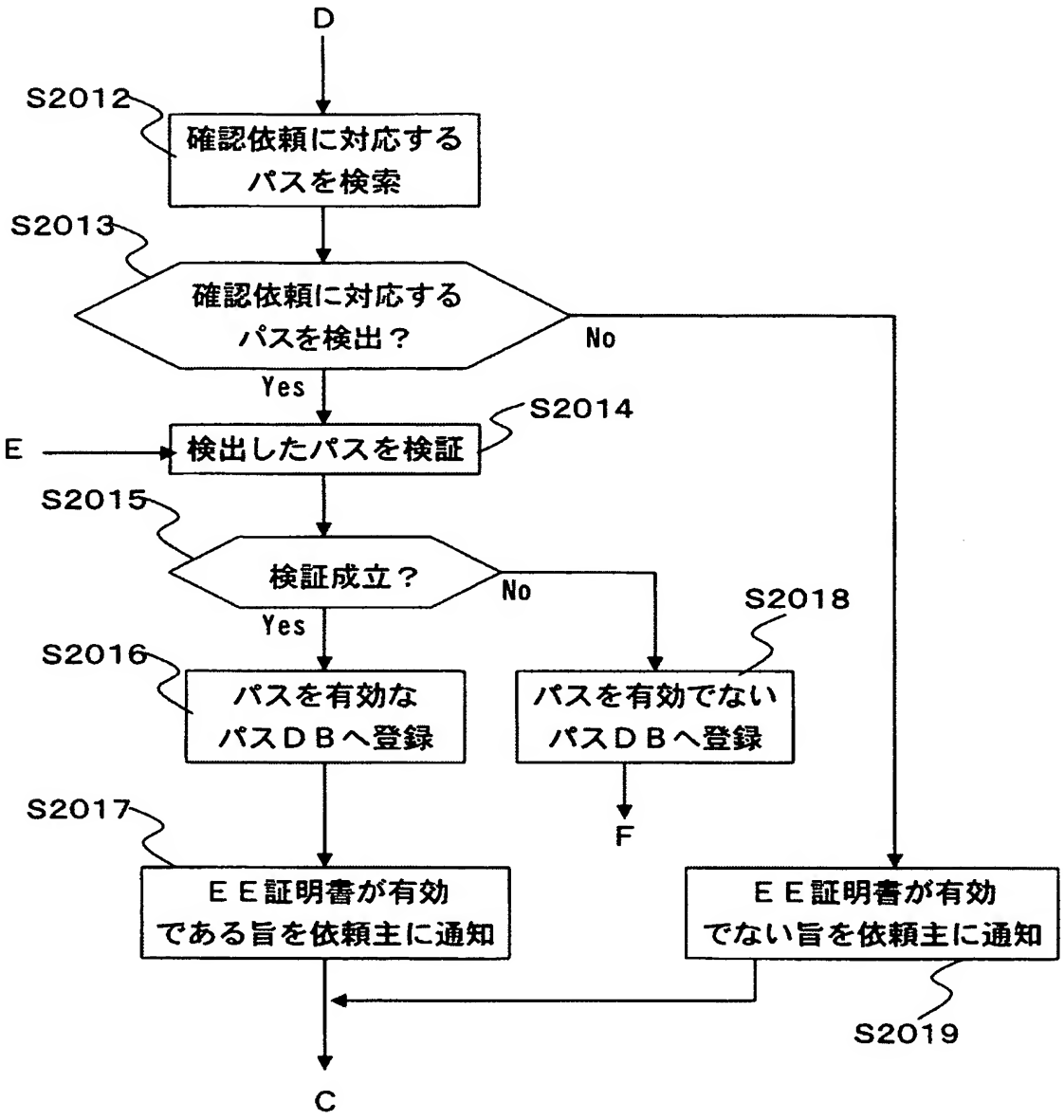
図１０　公開鍵証明書の有効性の確認動作



【図１１】

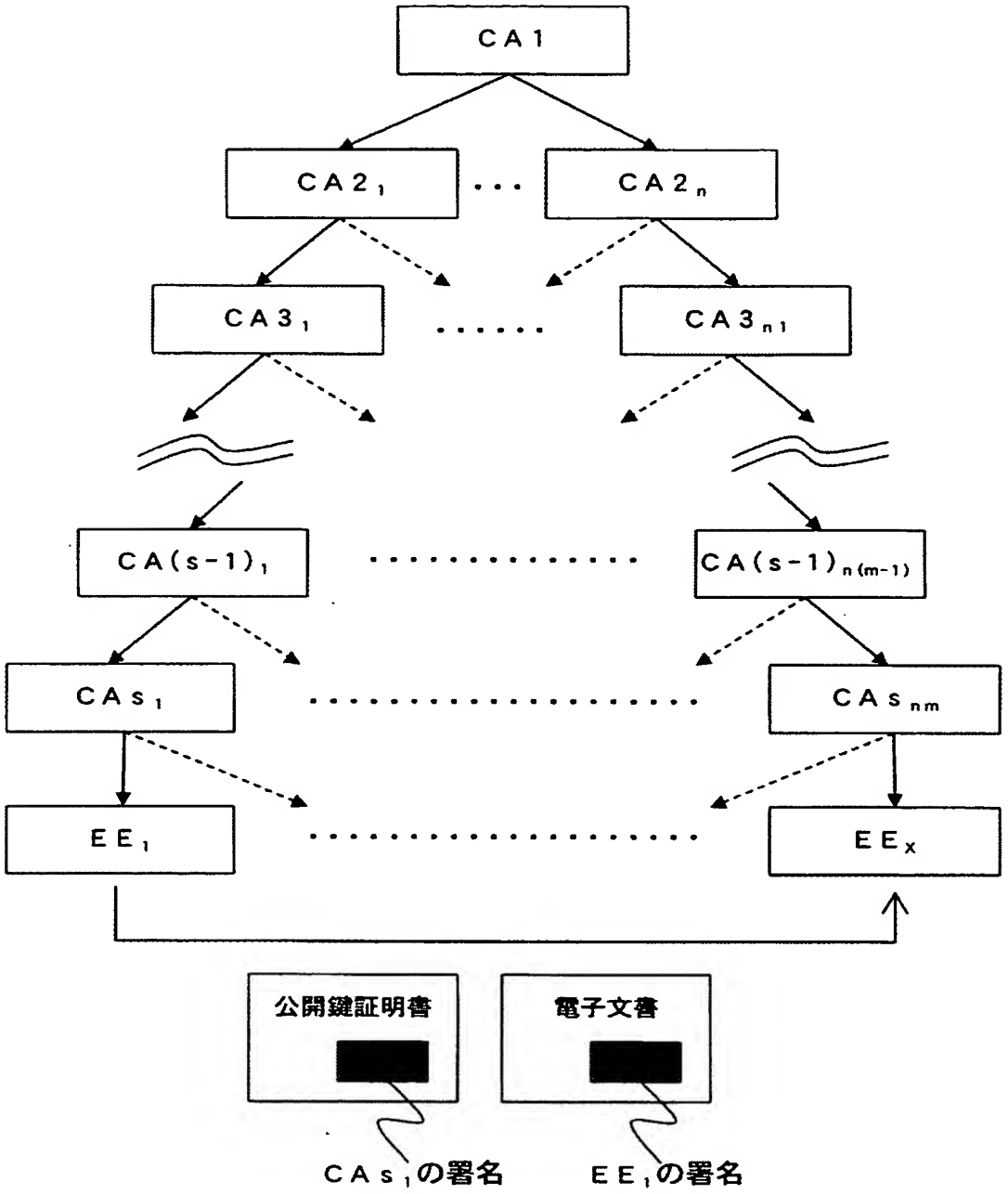
図１１

公開鍵証明書の有効性の確認動作



【図１２】

図１２


CA1
CA2₁
CA2ₙ
CA3₁
CA3ₙ₁
CA(s-1)₁
CA(s-1)ₙ₍ₘ₋₁₎
CAs₁
CAsₙₘ
EE₁
EEₓ
公開鍵証明書
電子文書
CAs₁の署名
EE₁の署名

【書類名】要約書
【要約】
【課題】
パス検索時以降に新たなパスが存在する場合でも適切な結果を応答し、かつ、エンドエンティティが公開鍵証明書の有効性確認を依頼してからその結果が分かるまでにかかる時間を短縮する。
【解決手段】
証明書の有効性確認局は、予め定期的に、パスおよび失効証明書リストを検索、検証し、当該検証結果に応じて、有効なパスおよび有効でないパスに分類して、データベースへ登録する。また、エンドエンティティから証明書の有効性確認依頼があった場合、当該依頼に対応するパスが、有効なパスデータベースか、有効でないパスデータベースのどちらに登録されているかを調べ、当該公開鍵証明書の有効性を判断する。一方、当該有効性確認依頼に対応するパスがデータベースに登録されていない場合には、新たにパス検索、検証を行うことにより、当該公開鍵証明書の有効性確認を行う。
【選択図】　図２

認定・付加情報

特許出願の番号	特願２００３－３５１５０９
受付番号	５０３０１６８９７０７
書類名	特許願
担当官	第八担当上席　　００９７
作成日	平成１５年１０月１４日

＜認定情報・付加情報＞

【提出日】　平成15年10月10日

特願２００３－３５１５０９

出　願　人　履　歴　情　報

識別番号　［０００００５１０８］

１．変更年月日　１９９０年　８月３１日
［変更理由］　新規登録
住　所　東京都千代田区神田駿河台４丁目６番地
氏　名　株式会社日立製作所